The Thesis Committee for Abdulmalik Alrasheed
Certifies that this is the approved version of the following Thesis

Inverse Problems in Drill-string Torsional Vibration

APPROVED BY
SUPERVISING COMMITTEE:

Kenneth Gray, Supervisor

John Jones

Inverse Problems in Drill-string Torsional Vibration

by

Abdulmalik Alrasheed

Thesis

Presented to the Faculty of the Graduate School of

The University of Texas at Austin

in Partial Fulfillment

of the Requirements

for the Degree of

Master of Science in Engineering

The University of Texas at Austin

August 2020

Acknowledgements

I would like to express my deepest gratitude to my academic advisor Dr. Kenneth Gray for his guidance through the completion of my Master's thesis. I am grateful for his support to pursue my research topic which allowed me to push the boundaries and tackle problems that I would not have been able to otherwise.

I have gained a lot of experience working and attending classes in different areas at UT which gave me the knowledge and experience to work on my research topic and for this I am very thankful. I am thankful for my colleagues for their help in all matters related to petroleum engineering that were new to me.

Warmest thanks go to my family and friend that supported me through this journey and made it possible through their love and encouragement.

Abstract

Inverse Problems in Drill-string Torsional Vibration

Abdulmalik Alrasheed, M.S. E.

The University of Texas at Austin, 2020

Supervisor: Kenneth Gray

In this study, a method is developed to better model drill-string torsional vibration by using data to calibrate a Partial Differential Equation (PDE) based model. Drill-string vibration is a complex phenomenon that is widely studied with several approaches to model the complexities encountered in real life. Sensors are now more widely available that can acquire high frequency data needed for the approach described in this study. The goal of this study is to use synthetic data to calibrate a PDE torsional model by using an inverse problem approach as a proof of concept to implementation on real data. The outcome of this approach is a calibrated model that can be used in control systems which can be implemented in the field to mitigate severe torsional vibration.

Torsional drill-string vibration was simulated using finite element method under different conditions of drill-string stiffness coefficients and damping coefficients varying along the entire length of the drill-string. Newmark beta method was used to perform the time stepping in the simulation giving us a more stable implicit formulation for time stepping which reduces the errors. Numerical methods were used to generate drill-string displacement data for the simulation time interval, which were then stored to act as input

for subsequent processing to simulate input data from sensors. Adjoint based method was used to calculate the gradients of the optimization problem. Using gradient descent, we incrementally update the parameters to better approximate the synthetic data until the original parameters were recovered.

Table of Contents

List of Figures

Chapter 1: Introduction and Research Objective

1.1 INTRODUCTION TO DRILL-STRING VIBRATION

Rotary drilling is the main method used to reach hydrocarbon bearing rock in the subsurface which facilitates the extraction of hydrocarbon resources. Forces are transmitted from the surface to the bottom of the hole through the drill-string which consists of tubular pipes connected together with a drill bit at the bottom. Increasingly, an electric Top Drive unit which is directly connected to the drill-string is used to generate the rotary motion and provide torque input to the drill-string. Down hole mud motors and rotary streerable systems (RSSs) are also commonly used, especially in deviated and horizontal wellbores. However, their additional cost for equipment and personnel can be significant.

The rotating motion of the drill bit causes the rock to fail creating cuttings and deepening the hole. Drilling takes place by applying the drilling parameters to the drill-string which are rotations per minute (RPM), weight on bit (WOB) and flow rate. The bottom of the drill-string is known as the Bottom hole Assembly (BHA) which is designed to suit the purpose of each drilling section, its main purpose in a vertical well is to provide the weight necessary to the bit to facilitate drilling. BHAs are a common failure point in drilling due to the high loads and vibration encountered.

While drilling, the drill-string acts as dynamical system subject to different perturbations from the drill bit interaction with the formation as well as drill-string collisions with the wellbore. These interactions cause excitations and oscillating behavior which can be very severe at certain drilling parameters. There are many factors that determine how the system behaves at different operating conditions such as mass, stiffness, damping and dynamic forces acting on the drill-string.

Severe drill-string vibration has been known for many years to be major cause of loss of drilling performance. The vibration negatively impacts the efficiency of the drilling process, reducing the rate of penetration (ROP). It can also cause damage to the cutting structure of the drill bit, necessitating a round trip to surface to change the drill bit. Vibrations can also cause severe damage through impact and cumulative fatigue damage to drill-string components such as the drill-collars (DC) and drill pipe (DP).

Conventional methods to mitigate the fatigue failure rely on limiting the operating hours of the bottom hole assembly (BHA) between inspections. But still, drill-string failure due to fatigue is a common occurrence leading to prolonged non-productive time (NPT) and financial losses. For these reasons, drill-string vibration has been a widely studied with the aim of finding the optimum conditions to avoid or minimize it.

1.2 MODES OF VIBRATION

There are three types of vibrations that occur during drilling operations: axial, lateral and torsional as shown in Figure 1-1. Axial vibration is the up and down motion along the longitudinal axis of the drill-string, usually resulting in a phenomenon called bit-bounce. Axial vibration is highly dependent on the bit interaction with the formation (Ghasemloonia, 2015). Lateral vibration is the sideways motion of the drill-string it manifests in the form of forward whirl or backward whirl. The major cause of lateral vibration is the drill-string being out of balance. Torsional vibration is the twisting motion of the drill-string which can be always present while drilling at various degrees of severity due to the great torsional flexibility of the drill-string at large depths. Actual drill-string vibration in the field can be very complex and these vibration modes can occur simultaneously (Saldivar, 2013).

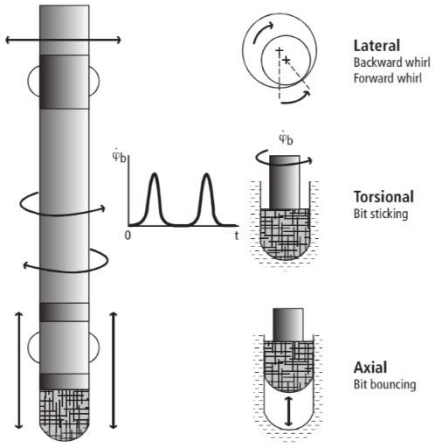


Figure 1-1: Lateral, Torsional and Axial vibrations. Source: Navarro-Lopez 2010

Stick-slip is a severe form of torsional vibration. This phenomenon happens when the drill bit bites into the formation and stops moving. Increasing torque from the rotation eventually causes the bit to break free and slip, then the process repeats. During the slip phase, the angular velocity of the bit can be several times greater than the angular rotation velocity of the drill-pipe at surface. When stick-slip occurs while drilling, oscillation in the drilling torque can be seen at surface as illustrated in Figure 1-2 below.

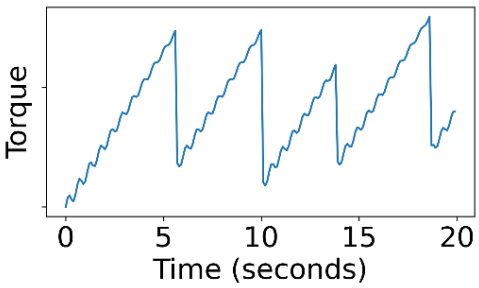


Figure 1-2: Illustration of torque behavior at surface under stick-slip conditions

Stick-slip can cause excitations at the drill bit that cause other forms of vibrations most commonly bit-bounce. Severe torsional vibration is regarded as a major cause of failure of the drilling assembly and hence is widely studied with the goal of avoiding stick-slip phenomenon (Saldivar, 2013). In this study, we will be mainly focused on torsional vibration behavior of the drill-string.

1.3 MODELING METHODS AND LITERATURE REVIEW

In this section we discuss different approaches to modeling vibrations in the literature.

1.3.1 Time Domain vs. Frequency Domain

There are two main categories of vibration models: frequency domain and time domain. Frequency domain models are a simple way to compute the natural resonance frequencies of the drill-string and BHA and have the advantage of being less computationally expensive than time-based methods. However, these methods have limitations as it can be difficult to incorporate time dependent behavior of the drill-string, and hence they may not reflect the true behavior of the drill-string. Frequency based methods are used for modeling and selection of BHAs and are used to determine operating conditions that avoid resonance. Zheng et al. used a frequency domain analysis of drill-string vibration to estimate fatigue damage (Zheng, 2017). Frequency-domain models are less useful for our approach in this study, since we would like to model the continually evolving dynamics of the drill-string.

The second type of models, time-domain models, are varying with time. The models are initialized at time zero, then the model is updated after a small time step Δt.

The time step is taken incrementally until the final time interval is reached to generate the time-dependent displacements of the drill-string. This approach is more powerful and will be the focus of this study since time-domain models are easier to integrate with time-based measurements from sensors in the field.

1.3.2 Torsional Models in Literature

Studies of drill string dynamics can be placed in several categories depending on the approach taken by the authors. One thing to consider is the type of vibration being studied. There are studies that attempt to separately model different modes of vibration including axial, lateral or torsional. Other studies consider the combined effect of two or more modes of vibration, also known as coupling (Ahmadian, 2007). Some studies focus on phenomena such as wellbore contact and drill bit/formation interaction (Ghasemloonia, 2015).

The mathematical modeling can be done in several ways. Some models utilize the lumped parameter model (Navarro-Lopez, 2007). The lumped parameter method simplifies modeling by describing sperate components of the drill-string as rigid masses connected with springs and dampers. Other models utilize continuous models by using equations that describe dynamic behavior of the drill-string in a continuous domain. Continuous models are then solved either by discretization or analytical approaches (Khuleif, 2008; Saldivar, 2013).

Yigit and Christoforou studied coupled interaction of torsional and lateral vibrations using lumped parameter method. Their model was derived for BHA motion only and assumes support at stabilizer positions. Input forces to the system were modeled as Hertzian contact to simulate the contact between the drill-string and wellbore. Stick-slip behavior was also investigated in their work (Yigit, 1998).

Navarro-Lopez et. al developed a torsional model with multiple degrees of freedom including nonlinear interactions. A sliding-mode control was proposed to suppress stick-slip. The system was discretized into four components: top-drive, drill-pipe, drill-collars and bit. These components were approximated as mass spring systems connected with additional damping forces. Simulations under different conditions were made, and the effect of WOB on the stability of vibration was studied (Navarro-Lopez, 2007).

Ahmadian et al. studied the motion of a drill string with coupling between axial, lateral and torsional vibrations. Complex interactions were considered in their model such as gyroscopic effects and wellbore interactions. It was shown that under certain ranges of operating conditions, resonance and whirling may occur. Contact forces were estimated through simulation. The fully coupled model was solved using Runge-Kutta method to generate the results. The authors suggest that the results exhibited chaotic behavior (Ahmadian, 2007).

Saldivar et al. studied torsional vibration and bit bounce by modeling the coupled axial and torsional dynamics. A wave PDE was used to generate the oscillatory behavior of the system. Lyapunov techniques were used to stabilize non-linear perturbations in the time-delay system. They analyzed the utilization of this method in control systems for suppression of torsional vibrations (Saldivar, 2013).

Khuleif et al. formulated a model using the Lagarangian approach for torsional/bending coupling and axial/bending coupling of vibration. The model accounts for gyroscopic and gravitational effects. Finite element method was used with the generalized eigen value to generate the dynamic equations of motion. In their work, the drill-string / borehole interaction was analyzed during short interval impacts (Khuleif, 2008).

1.4 MOTIVATION OF THIS STUDY

During drilling operations in various types of wells, such as deep hole or horizontal, the flexible drill-string is subject to various conditions that cause sever torsional vibration and stick-slip. There are complex non-linear interactions between the drill-string and formation as well as complex boundary conditions. Attempts to accurately reproduce these interactions resulted in increasing complexities such as coupling with axial vibrations or adding non-linear friction factor terms in the equation. However, these methods require exact modeling of the system to be accurate. But it can be very difficult to acquire an exact measurement of well conditions such as wellbore geometry, especially if the simulation is done ahead of drilling since there will always be deviation from the planned trajectory. Main factors causing limitation of current modeling techniques are the following unknowns.

- Drill bit / formation interactions.
- Drill-string / formation interactions.
- Damping due to drilling fluid.
- Borehole geometry.

For these reasons, complex models can be harder to use for active suppression of stick-slip. Some attempts at real-time torsional vibration control have been made using simplified models. One of the first systems for active control of torsional vibration due to stick-slip was developed by Shell Exploration (Javanmardi, 1992). This system was called Soft Torque Rotary System (STRS) and worked by making active adjustments to the electronic speed controller of the top drive system. It relied on measurement of electric current going to the drive motor to estimate the torque and use this measurement in the control loop (Javanmardi, 1992). Several field tests were performed since introduction of this system with varying degrees of success. In some cases, the STRS

significantly reduced torque fluctuations and stick-slip. However, the STRS is unable to suppress higher order modes of vibrations (Runia, 2013).

Since the introduction of early methods for torsional vibration suppression, the ability to acquire high frequency vibration measurements of the drill-string using MWD and wired drill-pipe have been developed. In this study, we propose a method to leverage data and increasing computational power to calibrate a torsional vibration method using data. The resulting calibrated model can replicate the phenomena occurring in real-life vibration while being simple enough to use for control purpose and analysis. This approach of model calibration is known as inverse problems and is widely used in Geophysical and other areas of study (Gladwell, 2005).

Calibration is key to creating an accurate dynamic vibration model. Instead of adding complexities to the model, we start with a generalized simple model that can be calibrated to match the measurement and the behavior observed in the field. Similar work was done by Dantas et. al on a lumped parameter model. They utilized the Cross-Entropy method, which is a stochastic algorithm, to calibrate a torsional dynamic model. The torsional model used by Dantas et al is based on constant stiffness, and up to six scalar parameters were optimized in their study (Dantas, 2019).

1.5 METHODOLOGY

1.5.1 Inverse Problem

In this study, we describe an approach of using inverse problem method to calibrate a PDE based model of torsional vibration. Here we will give a brief introduction of inverse problem and the terminology used.

Forward Model: A mathematical equation or process that generates data based on physical principles. Given a set of input parameters, such as material properties or geometry, it generates predicted data. Illustration given in Figure 1-3.

Inverse Problem: A mathematical equation or process that attempts to predict the parameters of a model based on data or measurements. Illustration given in Figure 1-4.

Model Parameters: The inputs to the model such as properties of material or forces acting on the drill-string. In this study, we will be mainly concerned with stiffness and damping coefficients. The inverse parameters of interest will be designated with the letter m when attempting to solve the inverse problem.

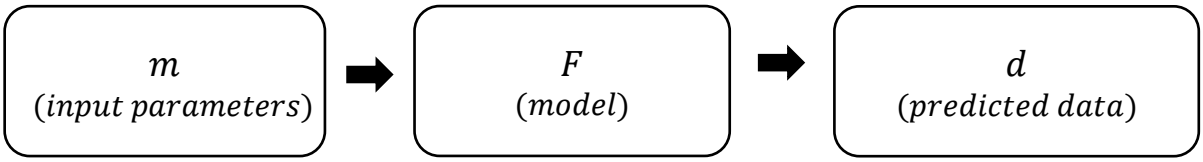


Figure 1-3: Forward Model

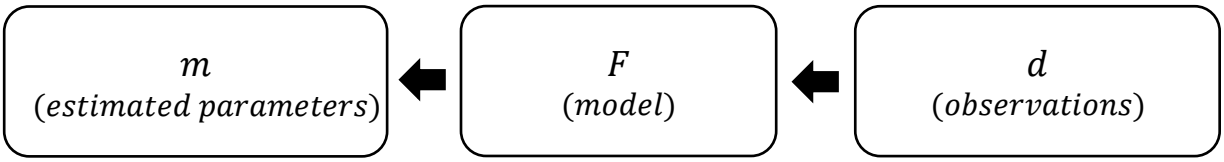


Figure 1-4: Inverse Problem

The Inverse problem relies on using data to recover the parameters. In this study we, will generate the synthetic using known parameters of stiffness and damping in the forward model. Inverse problems can be solved either deterministically or using probabilistic methods. Deterministic method will be used in this study as it is more developed with more resources in literature (Demanet, 2015).

1.5.2 Finite Element Method

Parameters and data in real life for a given system are typically continuous. In a metal rod, for example, material properties can be continuously varying along the entire length. To simplify these continuous models and allow us to compute the values, we need to convert these continuously varying values into discrete points that each have a unique value. Finite Element Method is one such method to convert the equation domain from continuous to discretized for solving purpose and will be used in this work.

The simulations made in this study were made by using the finite element library FEniCS in a Python programming environment. The FEniCS library was chosen since it provides tools for defining PDEs in terms that are very similar to their mathematical variational forms (Alnaes, 2015). The results of the simulation can also be accessed using other Python libraries. Data manipulation of vectors and arrays can be done in the same environment to achieve the goals of this study without the need to export and import results from an external software. The results were visualized using ParaView software and plots of data made using Matplotlib plotting library. Using these libraries and specifically written programming code, we were able to simulate the drill-string under various conditions and solve the for the gradient in the inverse problem.

1.5.3 Chapter Descriptions

In Chapter 2, we discuss the forward PDE based model and generate synthetic data using FEM discretization. In Chapter 3, the inverse model is developed and method of optimization using gradient descent is described. In Chapter 4 we apply the inverse model to the synthetic data and show the results compared to the true values used to generate the synthetic data.

Chapter 2: Torsional Vibration Model

Partial differential equations are used to model many physical systems such as membrane deformations or heat flow. Khuleif and Saldivar have shown the use of wave PDEs to model the torsional vibrations of a drill-string (Khuleif, 2008; Saldivar, 2013). In their work, they were concerned with coupling effects from other vibration modes and interaction effects. In this chapter, we address the formulation of a PDE torsional vibration model with known inputs such as stiffness, damping and forces, in order to generate synthetic data for the inversion process later.

2.1 PDE TORSIONAL MODEL

The drill-string is modeled as a rotating cylinder with variable stiffness, with damping and external forces acting along the entire drill string. It is assumed that the drill-string behaves in an elastic manner and the material is isotropic. It is also assumed that the behavior is the same whether the portion of drill-string is under tension or compression. The model is mathematically written as a hyperbolic partial differential equation.

$$I(x)\frac{\partial^2 u}{\partial t^2} - GJ(x)\frac{\partial^2 u}{\partial x^2} + D(x)\frac{\partial u}{\partial t} = f(x,t) \qquad \text{Eqn. 2-1}$$

This can also be written as

$$\frac{\partial^2 u}{\partial t^2} - k(x)^2\frac{\partial^2 u}{\partial x^2} + D(x)\frac{\partial u}{\partial t} = f(x,t)\,, where\ k = \sqrt{\frac{GJ}{I}} \qquad \text{Eqn. 2-2}$$

Where,

- u: $Angular\ displacement\ (rad)$
- x: $distance\ from\ bit$
- G: $Shear\ Modulus\ (N.m^{-2})$
- J: $Polar\ Moment\ of\ Inertia\ (m^4)$
- I: $Mass\ Moment\ of\ Inertia\ (kg.m^2)$
- k: Torsional stiffness $(N.m/rad)$
- D: $Damping\ coefficient\ (N.m.s/rad)$
- I. C. : $u(x,0) = 0$
- B. C. : $u(L,t) = 0$
- B. C. : $\frac{\partial u(0,t)}{\partial x} = 0$

The above PDE describes the dynamic torsional movement of the drill-string from the initial condition at t = 0 until the end of simulation at final time T. In the study by Saldivar et al. (Saldivar, 2013), a similar formulation was presented but with different boundary conditions to account for coupling effects. An illustration of this model is shown in Figure 2-1. The unknown quantity u stands for the angular displacement of the drill string and it is a function of two variables: location (x) and time (t). The symbol ∂ denotes the partial derivatives of u with respect to the different variables x and t. The term $f(x,t)$ represents the external forces acting on the drill string which are continuous and varying with time. The force input is represented as a steady state sinusoidal force that is applied at the bit and the response of the drill-string is measured as the system evolves.

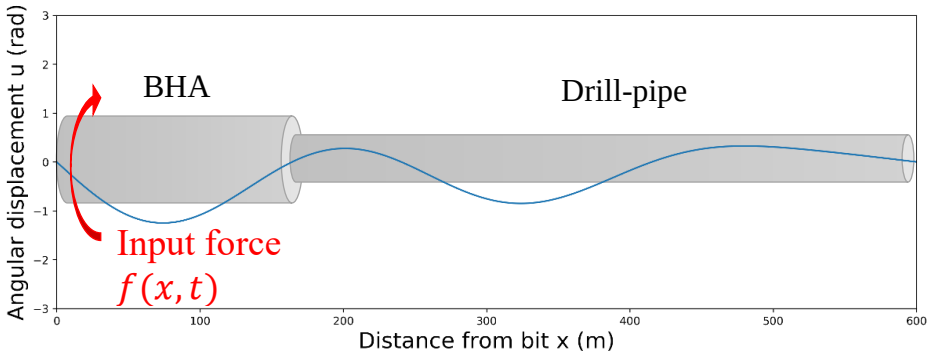


Figure 2-1: Illustration of the model with BHA and DP components. Displacement shown in blue line.

Equation 2-2 has the three parameters we are concerned with in this study, namely the torsional stiffness k, the damping coefficient D and the input forces $f(x,t)$. For simplicity, the formulation using the torsional stiffness k in the equation will be used. The

damping term D in this equation combines both the viscous damping from the mud and the dry friction between the drill-string and wellbore.

The model assumes constant rotation velocity at surface. All the results are relative to the rotating frame of reference equal to the rotating RPM of the top drive. The true angular velocity $\dot{\boldsymbol{\Omega}}_{\boldsymbol{True}}$ is the sum of the surface RPM and angular displacement velocity.

$$\dot{\boldsymbol{\Omega}}_{\boldsymbol{True}} = \dot{\boldsymbol{u}} + \dot{\boldsymbol{\Omega}}_{\boldsymbol{RPM}} \qquad \text{Eqn. 2-3}$$

In our case, the initial condition for the model is taken as static condition, i.e the string does not have any defection initially. However, a different initial condition can be substituted if known. Two boundary conditions are necessary to fully describe the motion, one at the bit and one at the rig floor. The boundary condition at the rig floor is a fixed boundary condition since the drill pipe is fixed to the top drive. The boundary condition at the bit is taken as a Neumann boundary which allows the free relative motion of the bit.

2.2 ENERGY ANALYSIS

Energy of the drill-string in a section or domain Ω is defined as the integral

$$E = \frac{1}{2}\int_{\Omega} I(x)(u_t)^2 + k(x)(u_x)^2 \, dx \qquad \text{Eqn. 2-4}$$

The first term in the integral is the kinetic energy term which depends on the velocity u_t at each point. The second term is the potential, or stored energy in the drill-string from the gradient u_x, which is the elastic energy based on Hooke's law from the twist in the tubular elements. Similar energy analysis in a discontinuous domain was presented by Khuleif et al. (Khuleif, 2008).

The power input into the system and dissipated energy due to viscous and frictional damping is given by the following:

$$P_{in} = \int f(t) \cdot u_t \, dt \qquad \text{Eqn. 2-5}$$

$$Energy\ Dissipation = \frac{1}{2}\int_{\Omega} D(x)\, {u_t}^2 \, dx \qquad \text{Eqn. 2-6}$$

When a force is applied at the bit, initially the energy of the drill-string increases until a balance point is reached. The energy will then oscillate around that balance point in case of stick-slip. Under such conditions, the energy input into the system and the energy dissipated in the form of viscous and frictional damping will balance out over long periods. Over shorter periods, however, energy fluctuation occurs in the form of stresses causing fatigue damage to the drill-string components.

2.3 FINITE ELEMENT METHOD IMPLEMENTATION

2.3.1 Introduction to FEM

The PDE torsional model described above is a continuous model for the drill-string. While there are some analytical methods to find the solution of such equations, these methods are only suitable for simpler models with constant coefficients.

In order to numerically solve the above PDE for torsional vibration of the drill-string, we will need to discretize the model in both spatial and time dimensions. The spatial discretization is handled by using the Finite Element Method (FEM). The FEM allows us to construct an approximation of the original equation using a predefined discretization of the equation domain. This discretization gives us a numerical system of equations based on stiffness, mass, damping and external forces acting on each 'element' which can be solved using numerical methods (Langtangen, 2019).

An advantage of FEM is that it simplifies handling of the boundary conditions and inherent geometric properties of the model (Langtangen, 2019). It also gives us freedom in our discretization so we can refine the mesh in areas where higher accuracy is required. Another advantage of the FEM is that it is well studied and there is great understanding of its implementations and the accuracy of the results. There are also several libraries to simplify the implementation of the FEM. A python library called FEniCS is one such library which is used in this work to define the drill string model and calculate the solutions (Alnaes, 2015).

The finite element method approximates the solution u by using basis functions ψ_i and multipliers c_i. These basis functions are chosen during the mesh selection and the unknowns c_i are what we solve for to find the solution to the differential equation. Illustration of this approximation is shown in Figure 2-2.

$$u = \sum c_i \psi_i(x)$$

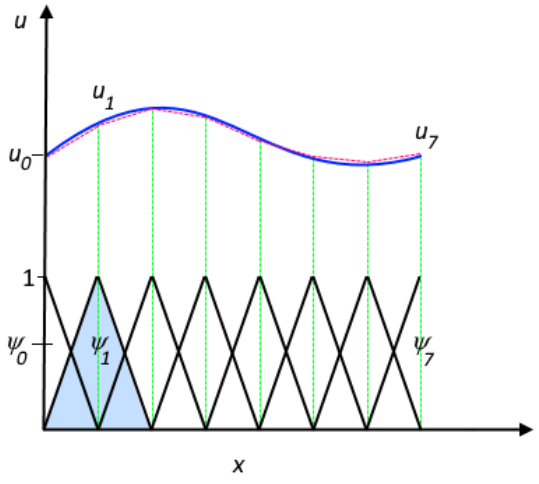


Figure 2-2: Illustration of FEM approximation (red dashed line) of true function (blue line) using basis functions. Source: (COMSOL., 2016)

For our work, we chose a second order Lagrange polynomial basis function for displacements u since it gives an adequate result without the computational complexity of higher order basis functions. The parameters k (stiffness) and D (damping) were chosen in the FEM model as a first order Lagrange element to reduce complexity and allow us to assign a single value for each element.

2.3.2 Variational Formulation

In order to solve the PDE ($-\nabla^2 u = f$) using FEM, where ∇^2 stands for the second order differential operator, we first need to convert this problem into a variational form. This is done by multiplying both sides of the equation by a test function v and integrating over the domain (Langtangen, 2019).

$$\int_{\Omega} -\nabla^2 u \cdot v \, dx = \int_{\Omega} f \cdot v \, dx$$

Then we use Green's rule to reduce the second order derivative of u into first order derivatives of u and v. In doing so, we need to take care of the resulting boundary conditions. If a fixed boundary condition is used, it can be eliminated from the formulation and can be incorporated in the domain definition when running the simulation. The resulting equation is

$$\int_{\Omega} \nabla u \cdot \nabla v \, dx = \int_{\Omega} f \cdot v \, dx$$

This is called the variation formulation, or weak form, of the original PDE instead of the strong form shown above. It is called the "weak form" since it no longer requires the existence of the second order derivatives of the function. Therefore, a discontinuous first order solution of the function is allowed in the weak form.

2.3.3 Variational Formulation of Torsional Model

We apply the previous procedure to derive the variation formulation, starting with Equation 2-2 we get

$$\int_{\Omega} (vu_{tt} - c^2 \nabla u \cdot \nabla v + Dvu_t)\ dx + \int_{\Gamma}\ vc^2 \nabla u \cdot n ds = \int_{\Omega}\ f \cdot v\ dx \quad \text{Eqn. 2-7}$$

This equation will be used to formulate the FEM solution where Ω is the domain of the equation and Γ is the Neumann boundary.

2.3.4 System of Equation

By using the above method for FEM discretization of the function u, the assembled equation of motion can be written as

$$M\ddot{U} + C\dot{U} + KU = F(t)$$

Where M is the mass matrix, C is the damping matrix, K is the stiffness matrix and F(t) is the external forces vector. $\ddot{U}, \dot{U}\ and\ U$ are vectors representing the acceleration, velocity and displacement, respectively.

2.4 Implicit Time Stepping Using Newmark Beta Method

Since this is a time dependent problem, we need to perform time stepping, i.e. going from displacements u_n at time t to displacements u_{n+1} at time t +Δt. There are several methods to perform this task. A simple method is the Euler method which approximates the second order time derivative by using the values at times t +Δt and t -Δt as follows.

$$\frac{\partial^2 u}{\partial t^2} = \frac{u_{n+1} - 2u_n + u_{n-1}}{\Delta t^2}$$

However, this results in an explicit time stepping which is inherently unstable. This means extremely small steps need to be taken in order to avoid instabilities and

errors which will increase computation time. Even with a small time step Δt, the errors will accumulate and the final result will slowly diverge from the correct answer.

There are several methods to solve this issue such as the generalized alpha method and different Runge-Kutta methods (Ahmadian, 2007). In this work, we used the Newmark beta method for time stepping to achieve a stable solution at larger time steps.

The Newmark method derives from the Taylor series expansion, where second order derivatives are approximated from accelerations at previous time step and current time step (Schöberl, 2016). Starting with equation of motion

$$M\ddot{U} + C\dot{U} + KU = f(t)$$

We start by writing the displacement and velocity at current time t +Δt in terms of acceleration at current step and displacement, velocity and acceleration from previous step.

$$U_{n+1} = U_n + \Delta t\dot{U}_n + \frac{\Delta t^2}{2}((1-2\beta)\ddot{U}_n + 2\beta\ddot{U}_{n+1}) \quad \text{Eqn. 2-8}$$

$$\dot{U}_{n+1} = \dot{U}_n + \Delta t((1-\gamma)\ddot{U}_n + \gamma\ddot{U}_{n+1}) \quad \text{Eqn. 2-9}$$

Solving and rearranging to in terms U_{n+1} and U_n we get

$$AU_{n+1} = B_n$$

Where

$$A = \frac{M}{\beta\Delta t^2} + \frac{\gamma C}{\beta\Delta t} + K \quad \text{Eqn. 2-10}$$

$$B_n = f(t_{n+1}) + M\left[\frac{U_n}{\beta\Delta t^2} + \frac{\dot{U}_n}{\beta\Delta t} + \left(\frac{1}{2\beta} - 1\right)\ddot{U}_n\right].. \\ +C\left[\frac{\gamma U_n}{\beta\Delta t} - \dot{U}_n\left(1 - \frac{\gamma}{\beta}\right) - \Delta t\left(1 - \frac{\gamma}{2\beta}\right)\ddot{U}_n\right] \quad \text{Eqn. 2-11}$$

To solve these equations, at each time step we solve for U_{n+1} then using Equations 2-8 and 2-9 we find velocity and acceleration and use those values to solve for next time step (Lindfield, 2019). This gives an implicit time stepping method which is inherently more stable and gives better accuracy results at smaller Δt.

The Newmark method satisfies energy conservation for correctly chosen parameters γ and β. Depending on choice of parameters β and γ we get:

- $\gamma = \frac{1}{2}$: Conservation
- $\gamma > \frac{1}{2}$: Damping
- $\gamma < \frac{1}{2}$: Unstable
- $\gamma = \frac{1}{2}$, $\beta = \frac{1}{4}$ Unconditionally stable. This is called the average acceleration method and are the values used for this work (Schöberl, 2016).

2.5 Error Analysis

Since these discretization methods are only an approximation of the true PDE, there will be an error which is the difference between true displacements and the approximation. We would like to get an assessment of the amount of error and optimize the spatial and time discretization to reduce the error without unnecessarily increasing computation time.

2.5.1 Spatial Discretization

The spatial discretization or mesh size is the measure of the size of each element in the model Δx

$$\Delta x = \frac{L}{N}$$

where L is the length of the drill-string and N is the number of elements. As we decrease Δx the error will decrease to better approximate the true solution. However, we are unable to compare the error to the analytical solution since the problem does not have a straightforward analytical solution for variable stiffness and damping such as in our case. Instead, we approximate the true solution by using a very fine mesh N = 10,000

then we compare the different meshes results with this solution. The relative error at mesh size N is

$$e_N = u_{10,000} - u_N \qquad \text{Eqn. 2-11}$$

The torsional model was solved for different mesh sizes and the errors calculated. Errors were plotted in log-log scale vs the mesh size shown in Figure. From this analysis we chose mesh size of 1,000 elements (Δx = 0.6 m) since it has sufficiently low error.

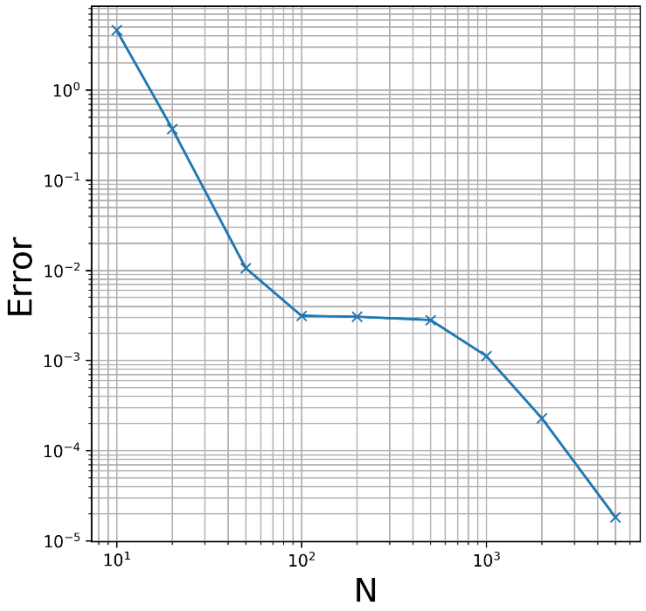


Figure 2-3: Relative errors at different mesh refinements N

2.5.2 Time Discretization:

Similarly, for time discretization, we analyze the error based on the size of time step Δt.

$$\Delta t = \frac{T}{N_t}$$

Where T is total simulation time and N_t is the number time steps in the simulation. We calculate the relative error by comparing to the solution at N_t = 4,000 (Δt = 0.0025).

$$e_{N_t} = u_{4,000} - u_{N_t} \qquad \text{Eqn. 2-11}$$

The equation was solved for time step sizes Δt and the error calculated. Errors were plotted in log-log scale vs number of time steps shown in Figure. From this analysis we chose a time step Δt = 0.005 (N_t = 2000).

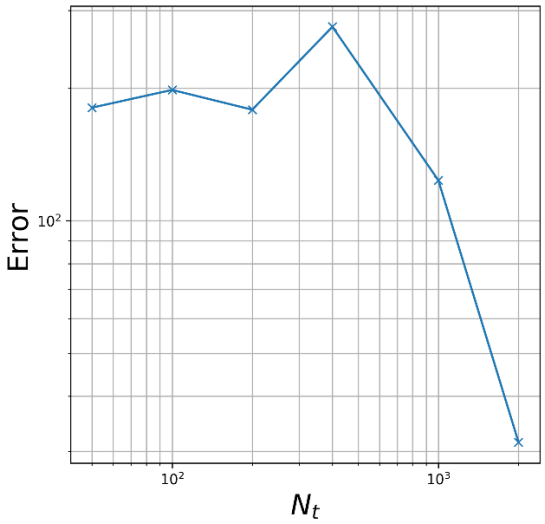


Figure 2-4: Relative errors at different number of time steps N_t

2.6 SIMULATION RESULTS

Simulation of the torsional model was made by using the Python library FEniCS. Different parameters of stiffness and damping were chosen to see the system response. At each simulation interval, only one parameter was changed such as the input frequency, stiffness and damping while the rest of the parameters were held constant. The base parameters used in the simulation are shown in Table-1 below.

Parameter	Description	Value	Unit
L_{DP}	Length of Drill Pipe	500	m
L_{BHA}	Length of BHA	100	m
k	Torsional Stiffness	500 - 750	$N.m/rad$
D	Damping Coefficient	0.05 – 0.8	$N.m.s/rad$
$\boldsymbol{\rho}$	Density of Material	7800	kg/m^3
$\boldsymbol{f}$	Input Force	1500 - 3000	$N.m$

Table 1: Numerical values of parameters used in simulation

The result of the simulation is displacement data at 2000 time steps. Paraview was used to visualize the transient system response by importing the simulation results into the software. A snapshot of vibration displacement at different frequencies is shown in Figure 2-5. The actual visualization in the software shows the time evolution of the system instead of just the snapshot shown.

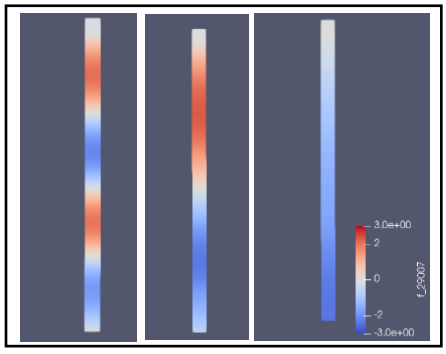


Figure 2-5: Snapshot of Paraview visualization under different input force frequencies

To show results of vibration simulation, plots were made of the displacement data. The plots shown are simplified by either showing displacements at a single point through the simulation, usually at the bit, or showing all the displacements of the drill-string at one time step. Simulation was done using different damping coefficient distributions shown in Figure 2-6 sample of the results is shown in Figure 2-7.

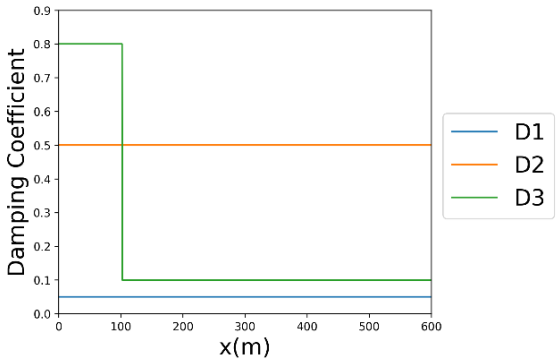
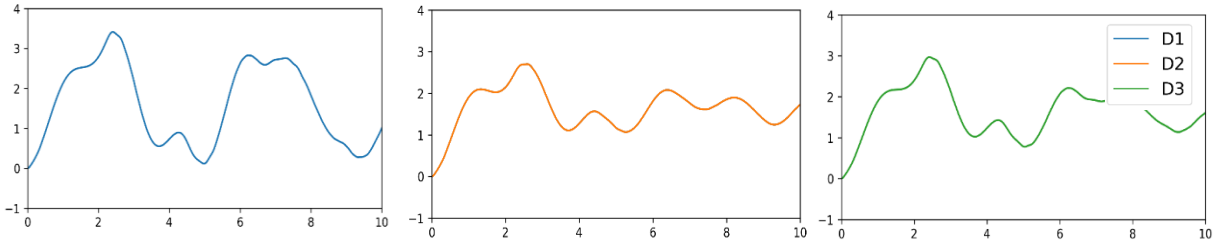


Figure 2-6: Different Damping Coefficients



Time (seconds)

Figure 2-7: Displacements at bit using the different damping coefficients shown in Figure 2-6

To create a more representative synthetic data to use for the inverse problem later, we added noise to the displacements results from the simulation. A random normal distribution of noise was added at each point with a mean of zero and standard deviation of 0.1 units. The result is shown in Figure 2-8.

$$u_d = u + N(0, 0.1) \qquad \text{Eqn. 2-12}$$

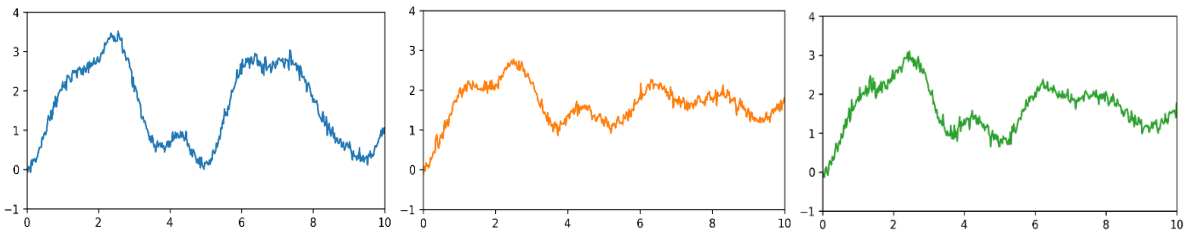


Figure 2-8: Synthetic data generated by adding noise to simulation results

Chapter 3: Inverse Torsional Vibration Model

3.1 INTRODUCTION

Determining model parameters from observed data is an important task that is encountered in many areas of study. To achieve this task, inverse problem formulation is used to find the unknown parameters of the model. The formulation of the problem involves creating a least squares misfit between synthetic and observed data. Then we attempt to minimize this misfit by varying the parameters until a minimum of the function is found. Methods such as gradient descent, conjugate gradient and Newton's method are used to find the minimum of the function. However, the minimum that is found could be completely different from the actual parameters from our observed data. This is due to ill-posedness of the problem, where the solution to our inverse problem may not be unique, stable or exist at all.

The inverse method approach has gained a lot of attention recently with applications in geophysical exploration (Plessix, 2006). Advancements in computation power and algorithms that allow faster results helped wider adoption in various fields. There is a growing literature covering the topic with many publications regarding the mathematical formulation (Demanet, 2015). Few papers have discussed the application of the inverse problem method in drill string vibration application. One work in this area was done by Dantas et. al, where they used an inverse problem approach to find six scalar parameters to fit a torsional model (Dantas, 2019). There are several books and papers that discuss inverse wave propagation (Gladwell, 2005). In the following sections we formulate the inverse problem of the torsional PDE model and give expressions and an algorithm for the solution. In the next chapter, we present the numerical result of the inverse problem.

3.2 LEAST SQUARES MISFIT FORMULATION

Our goal is to find the parameters (m) that, when used in the PDE, matches the observations (u_d). This is can be done by minimizing a least-squares data misfit between the predictions and observations:

$$\min_m \boldsymbol{\Phi}(m) := \frac{1}{2}\int_0^T \int_\Omega (u - u_d)^2 dxdt \qquad \text{Eqn. 3-1}$$

Where u is calculated from the forward PDE model depending on the input parameter m.

$$u_{tt} - m\Delta u + Du_t = f$$

Here we are solving for the squared stiffness k^2. Hence, we substitute it with unknown parameter m. Equation 3-1 represents the squared difference at each point on the drill-string between prediction and observation over the period of simulation [0,T].

3.3 ILL-POSEDNESS

Unfortunately, the above formulation of the inverse problem cannot be directly solved due to ill-posedness. In mathematics a problem is said to be well-posed if it satisfies these three conditions

1. Existence: There is at least one solution to the problem.
2. Uniqueness: There is only one solution.
3. Stability: Small changes in the input parameters lead to small changes in the solution.

Inverse problem of the torsional vibration PDE is ill-posed since it does not satisfy all the above conditions, particularly that the answer is not unique. Thus, the problem cannot be solved directly because the solution would be unstable and may give

false or physically impossible answers such as negative stiffness. This problem is exacerbated by noise in the data. If we attempt to solve for the parameters in noisy data, the solution would be corrupted by the noise in the data. The resulting solution would try to approximate the noise in the data which is not representative of the true model parameters.

There are several ways to deal with ill-posedness (Gladwell, 2005). A common method is regularization, which is equivalent to adding a penalty term to the data misfit term to help stabilize the solution. The regularization helps in smoothing the curvature of the minimization function, thereby ensuring a stable unique solution can be found regardless of noise in the data.

3.4 REGULARIZATION

We add the regularization term to the data misfit in Equation 3-1 in order to have a stable inverse problem formulation.

$$\min_m \boldsymbol{\Phi}(m) \coloneqq \frac{1}{2}\int_0^T \int_\Omega (u - u_d)^2 dxdt + \frac{\beta}{2}\int_0^T \int_\Omega \nabla m \cdot \nabla m \, dxdt \qquad \text{Eqn. 3-2}$$

Here, β is the regularization parameter which determines the magnitude of the regularization. A careful choice of the regularization parameter should be made such that the solution is not over regularized. By doing so, the information that can be extracted from the original data will be reduced. There are several methods to find optimal regularization parameter such as the Morozov's discrepancy principle (Demanet, 2015). By trying different values, we can choose a value that stabilizes the solution without eliminating too much information from the observed measurements. This can also be done by observing the magnitude of the data misfit term and comparing it to the

magnitude of the regularization term while making sure one term does not overpower the other.

The regularization operator requires some insight on how the true inversion parameter looks. In this study we use the L-2 norm Tikhonov regularization on the inversion parameter m. This regularization method imposes a penalty on the magnitude of the gradient of the parameter field. With this type of regularization, the effect is smoothing of the parameter field and eliminating highly oscillatory terms. The downside of such regularization is that it tends to smooth out discontinuities inherent to the parameter field, such as at the interface between drill collars and drill-pipe. However, this assumption of smoothness is accurate in the majority of the parameter field along the drill-string. With this regularization the problem is well-posed, and a solution can be found for the inverse vibration problem.

3.5 Optimization Approach

We would like to minimize the function described in Equation 3-2, sometimes called the cost function or the loss function, for the unknown parameters m_i in order to fit the data. This is an optimization problem in high dimensions, as many dimensions as there are parameters to optimize, and there is a global minimum of the function that we would like to find. A characteristic of a minimum of the function is having the first derivative of the function be equal to zero at that point. The optimization space is a subset of Euclidean space $\mathbb{R}^{\boldsymbol{n}}$, where n is the number of variables m_i in our optimization parameter field, known as the search space.

If the optimization problem is convex, i.e. has a positive second order derivative everywhere, there will be one unique global minimum. Some inverse problems are non-

convex, in such case there will be one or more local minima that are greater than the true global minimum. Local minima can cause an issue for algorithms developed to solve convex problems. These algorithms will quickly converge to the local minimum and become stuck, not being able to find the global minimum. Some methods have been developed to solve non-convex optimization. In our case we will treat the optimization problem as sufficiently convex given enough regularization and will be using methods developed for convex optimization.

The optimization problem often has a set of constraints that must be satisfied. In our case the constraint is that displacements u must satisfy the torsional vibration PDE that we specified. A constrained optimization specifies the constraint condition externally to the cost function. Finding a solution that both satisfies the constraint condition and minimizes the function, is often a very difficult problem. To fix this issue, we would like to convert our constrained problem into an unconstrained optimization problem. Unconstrained optimization has many efficient algorithms that use derivatives to find global minima. In an unconstrained problem we can analytically compute terms for the derivates that help us solve the problem.

3.6 LAGRANGIAN FORMULATION

To convert our problem from a constrained optimization to an unconstrained optimization we use the method of Lagrange multipliers (Demanet, 2015). This method allows us to solve the constrained optimization problem without explicit parametrization in terms of the constraint. It works by converting the problem into what is called the Lagrangian function by adding the condition term multiplied by an unknown variable λ. The solution of the original problem corresponds to the stationary points of the Lagrangian function with respect to the original parameter and the additional term λ.

$$\min_{m} \boldsymbol{\Phi}(m)$$

Subject to constraint $G(m) = 0$

The Langrangian can be written as

$$\boldsymbol{\mathcal{L}}(m, \lambda) = \boldsymbol{\Phi}(m) + \lambda G(m)$$

Now that we have the Lagrangian function in terms of the variables m and λ, we can find the solution by setting the various partial derivatives equal to zero.

$$\boldsymbol{\nabla \mathcal{L}}(m, \lambda) = \mathbf{0}$$

For our formulation we will use the symbol p for Langrange multiplier since this term represents the adjoint variable which will be used for calculation of the gradient later. To solve the inverse problem of vibration PDE, we cannot directly calculate the parameters by setting the gradient equal to zero. Instead, we will use the derivatives of the Lagrangian function to calculate the gradient of the function and update the solution.

Applying the above described method, we have the following expression for the Lagrangian formulation of the inverse problem.

$$\boldsymbol{\mathcal{L}}(u, m, p) = \tfrac{1}{2}\int_0^T \int_\Omega (u - u_d)^2 dxdt + \frac{\beta}{2} \int_0^T \int_\Omega \nabla m \cdot \nabla m dxdt + \int_0^T \int_\Omega [pu_{tt} - m\nabla u \cdot \nabla p + Dpu_t - pf] dxdt - \int_0^T \int_\Gamma pc^2 \nabla u \cdot n ds dt \quad \text{Eqn. 3-3}$$

3.7 OPTIMIZATION ALGORITHMS

To solve the optimization function, we would like to find the combination of parameters that minimizes the function. If we can know the value of the function at every possible combination in parameter space, we can simply pick the lowest value. Unfortunately, this is computationally infeasible as it would take an inordinate amount of time to calculate the loss function at every possible point, not to mention the extreme memory requirement for such computation. Luckily, there are several algorithms developed to solve such problems that can do so much more efficiently. These methods usually start by making an initial guess of the unknown parameters and calculating the cost function at that point. Then, the goal of all the algorithms is to move from the current point to a point that is closer to the global minimum.

Algorithms to solve optimization problems are classified based on the order of the derivative of the cost function that is used in the algorithm. First order methods, such as gradient descent, use the first order derivative to find the gradient at the current position in parameter space. The direction of the gradient is the direction with maximum change locally and can be used to minimize the function. But these methods can require larger number of steps to converge. Gradient descent was used in this work and is discussed in more detail in the next sections.

Second-order algorithms such as Newton's method converge in a lower number of iterations compared to first-order methods. These methods create a local quadratic approximation of the objective function, giving better understanding of the shape of the function. However, the trade-off is the additional work needed to calculate the second order derivatives. This can be very complicated because, while gradient is represented as a vector, the second order derivate is a matrix known as the Hessian that is much harder to compute than the gradient.

3.8 GRADIENT DESCENT

Gradient descent is an optimization algorithm that aims to minimize the cost function by heading in the direction of steepest descent from the current position. For illustration purpose a simple objective function $F(m)$ with two parameters m1 and m2 is shown in Figure 3-1. The steepest descent direction is perpendicular to the contour line at the starting point P as shown represented by the arrow in the figure. This steepest descent direction is the negative of the first order derivative of the cost function. The gradient of the function g

$$g = \nabla F(m)$$

for our simple case this becomes the vector of partial derivates of the parameters.

$$g = \begin{bmatrix} \frac{\partial F}{\partial m1} \\ \frac{\partial F}{\partial m2} \end{bmatrix}$$

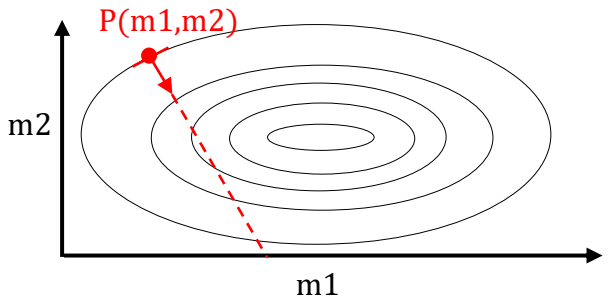


Figure 3-1: Gradient and Line search for a function F(m) with two parameters

The gradient gives us the direction to find the next point P_{n+1} by subtracting the gradient from the previous point. However, as seen in Figure 3-1, the cost function initially decreases in the direction of g reaching a minimum then it increases again. For

this reason, we add the scalar parameter α to represent the step length. The step length is adjustable and can be changed in each iteration.

$$P_{n+1} = P_n - \alpha g \qquad \text{Eqn. 3-4}$$

For sufficiently small step length α, the reduction of the cost function is guaranteed. However, if α is very small, it will take an exceedingly large number of iterations for the algorithm to converge to the optimum point. Methods to find the optimum step length α , known as line search methods, are available. In this work, a backtracking line search method is used. In this method, a large step size is used initially, and the cost function is evaluated at this new point, then the step size is recursively reduced until a sufficient decrease of the cost function is observed.

The method of gradient descent can suffer from phenomena called zigzagging shown in Figure 3-2. This happens because the gradient only informs minimization direction local to the point. As we head in the gradient direction, we continually overshoot the direction to the true minimum of the function. This is exacerbated by the stretching of the parameter space, where the gradient is stronger in certain directions and weaker in others. The result is slow convergence of the optimization, since the number of steps required can increase exponentially as we approach the minimum.

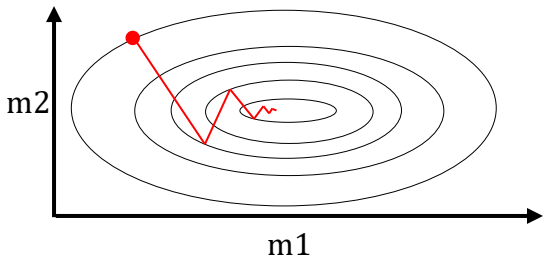


Figure 3-2: Zigzagging behavior of gradient descent optimization

3.9 ADJOINT METHOD

To find the solution, we want to find the gradient $\boldsymbol{\partial \Phi / \partial m}$**.** However, m here has a different value at each element and it would be inefficient to calculate the gradient by taking the derivative of the equation with respect to each parameter m_i. Adjoint methods provide us with an efficient way to evaluate the gradient with respect to parameters of a PDE (Johnson, 2012). This method is efficient since it reduces the computational cost of calculating the gradient and can be as low as a single solve of the forward equation for each gradient evaluation. This is especially important as the number of parameters increase (Demanet, 2015).

3.10 IMPLEMENTATION FOR INVERSE TORSIONAL VIBRATION

To find the adjoint and gradient formulation of the inverse vibration model, we start with the Lagrangian

$$\boldsymbol{\mathcal{L}}(u,m,p) = \frac{1}{2}\int_0^T \int_\Omega (u - u_d)^2 dxdt + \frac{\beta}{2}\int_0^T \int_\Omega \nabla m \cdot \nabla m dxdt$$
$$+ \int_0^T \int_\Omega [pu_{tt} - k^2 \nabla u \cdot \nabla p + Dpu_t - pf]dxdt - \int_0^T \int_\Gamma pc^2 \nabla u \cdot n ds dt$$

If we are interested in finding an inverse solution for stiffness, we replace k^2 with m. Similarly, for damping inversion we replace D with m. Then we take the variation $\delta_m \boldsymbol{\mathcal{L}}(u,m,p)$ to find the gradient.

Gradient Expression for Stiffness:

$$G(m,\hat{m}) = \beta \int_0^T \int_\Omega \nabla m \cdot \nabla \hat{m} dxdt + \int_0^T \int_\Omega \hat{m}\, \nabla u \cdot \nabla p \, dxdt \quad \text{Eqn. 3-5}$$

Gradient Expression for Damping:

$$G(m,\hat{m}) = \beta \int_0^T \int_\Omega \nabla m \cdot \nabla \hat{m} dxdt + \int_0^T \int_\Omega \hat{m}\, p \cdot u_t \, dxdt \quad \text{Eqn. 3-6}$$

The adjoint equation is found by taking the variation $\delta_u \mathcal{L}(u, m, p)$. The result is converted to strong form to get the following adjoint PDE for variable p

$$p_{tt} - k^2 \Delta p - D p_t = -(u - u_d) \qquad \text{Eqn. 3-7}$$

- Terminal Condition : $p(x, T) = 0$
- B. C. : $p(L, t) = 0$
- B. C. : $\frac{\partial p(0,t)}{\partial x} = 0$

The adjoint equation is similar to the forward PDE except that the damping term has changed sign. The right-hand side of the equation has the residual term (u_d – u) which goes to zero as the predictions get closer to the observations. Unlike the forward problem, this is a terminal condition equation since we start at final time T and time stepping is done backwards. Similar results were shown by Plessix for inversion of waves for geophysics applications (Plessix, 2006).

3.11 STEPS FOR SOLUTION

To solve the inverse problem, we use the following steps to calculate the gradient and update the parameter values.

1. Make an initial guess of the unknown parameters m.
2. Solve the forward problem for u for all time steps using m and store values.
3. Solve adjoint equation 3-7 for p in all time steps using the data residual (u_d – u).
4. Calculate the gradient using equation 3-5 or 3-6 depending on the parameter of interest.
5. Use a line search algorithm to update the parameters in the direction of the gradient.
6. Repeat steps 2-5 until convergence is achieved.

Chapter 4: Results and Discussion

The above inverse problem procedure was used to recover the parameters used when the synthetic data was generated as described in Chapter-2. The procedure was applied to two cases. In the first case, we attempted to find the stiffness while keeping the remaining parameters fixed. In the second case, our goal was to find the damping coefficients while assuming stiffness is known.

4.1 Numerical Result for Stiffness Parameter

To find the stiffness parameters, we start with a guess for the value of stiffness assuming it to be constant along the length of the drill-string. We iteratively calculate the gradient based on the algorithm discussed in section 3.11, and the values are updated. Figure 4-1 shows the process as we approach convergence to the true value. Due to slow convergence rate of first order optimization used, it takes increasing larger number of iterations as we approach the correct value. For this reason, the results shown were terminated prior to reaching complete convergence due to extremely slow convergence.

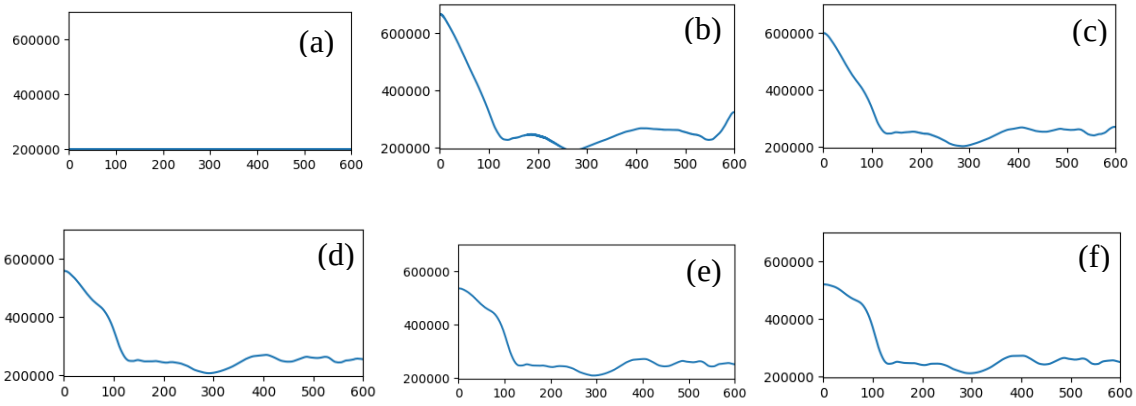


Figure 4-1: Value of recovered parameters at different iterations from initial guess (a) to final result at (f)

The final recovered parameters compared to the ground truth used to generate the synthetic data is shown in Figure 4-2. It can be seen from the figure that the resulting inversion parameters fluctuate around the true value from the original simulation. Due to regularization, the discontinuity at the transition from DP to BHA is hard to recover for the exact true values with the chosen inversion process. A major reason for this is the smoothing effect from the regularization operator used.

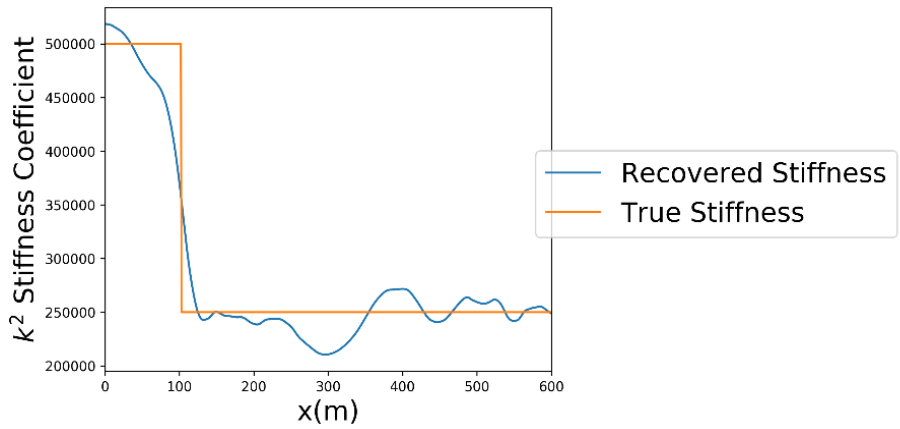


Figure 4-2: Comparison of recovered stiffness value with ground truth.

The resulting inversion parameters were used to generate displacement data for the entire simulation interval [0, T]. A comparison of the generated data with the original synthetic data is shown in Figure 4-3. The result shows a good match of the prediction to the true values even though the data is noisy. The error was calculated with increasing time as shown in Figure 4-4. There is a very good match to observations initially, the error then increases and fluctuates for the remaining length of the simulation.

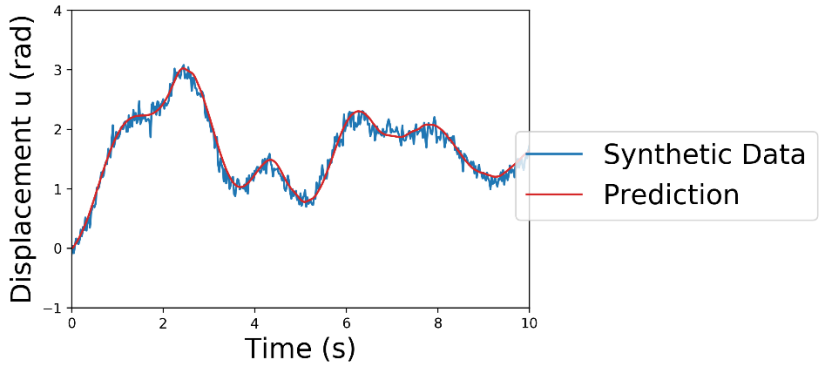


Figure 4-3: Prediction vs. True values

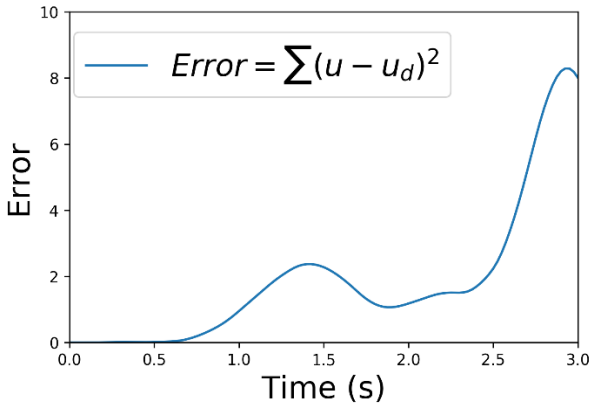


Figure 4-4: Squared Error between observations and prediction for first 3 second of simulation time.

4.2 Numerical Result for Damping Parameter

Similarly, we apply the process using the gradient terms derived for damping to recover the damping coefficients. The results for two different damping coefficient distribution is shown in Figures 4-5 and 4-6. For case-1 we start with a guess value of 0.5 Nms/rad and 0.4 Nms/rad for case-2.

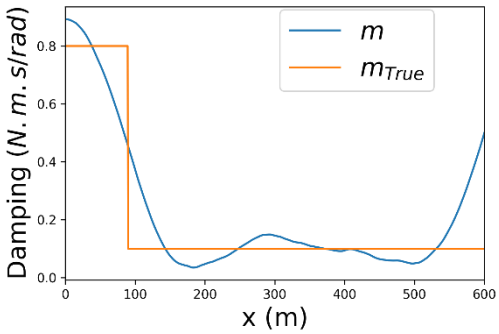


Figure 4-5: Recovered Damping Coefficients compared to true value Case-1

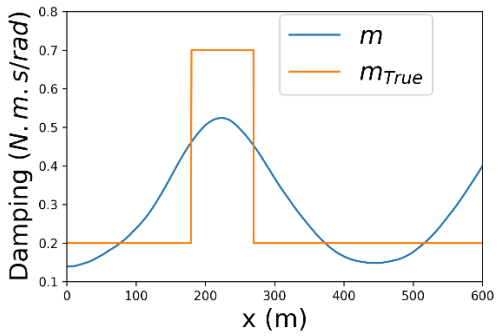


Figure 4-6: Recovered Damping Coefficients compared to true value Case-2

The recovered damping coefficients are smoothed out even more due to their lower effect on displacement compared to stiffness. As we optimized the loss function, data misfit becomes very low and converges even though damping coefficients still do not match the true values. Another effect which can be seen in the recovered parameters is that the error is greater at the fixed boundary condition on the right. This is because damping has less effect in that area due to the fixed condition compared to the greater velocities reached further away which increase the damping effect on the loss function.

4.3 NOTE ON GRADIENT CALCULATION

When attempting to calculate the gradient of the loss function, it was observed that the function is highly non-linear. The gradient calculation was unstable especially when the initial guess values were very far from the true value. Attempts to calculate the gradient using all the displacement values at all time steps did not converge to the true values. This is due to the greater divergence of the predictions u compared to u_d as simulation time increases for the conventional approach shown Figure 4-7.

$$u_{d0} \rightarrow u_{d1} \rightarrow \cdots\cdots\cdots \rightarrow u_{dn-1} \rightarrow u_{dn}$$

$$u_0 \rightarrow u_1 \rightarrow \cdots\cdots\cdots \rightarrow u_{n-1} \rightarrow u_n$$

Figure 4-7: Conventional approach for calculating residual ($u - u_d$) for all time steps

To solve this, we did the simulation and adjoint calculation for a shorter interval initially. Gradually as the parameter converges, we increase the simulation interval which helped stabilize the problem as shown in Figure 4-8, where the symbol s stand for the shorter interval length. However, this procedure takes a longer time to converge.

$$u_{\mathrm{d0}} \rightarrow u_{\mathrm{d1}} \rightarrow \cdots \rightarrow u_{\mathrm{ds}}$$

$$u_0 \rightarrow u_1 \rightarrow \cdots \rightarrow u_{\mathrm{s}}$$

$$u_{\mathrm{d0}} \rightarrow u_{\mathrm{d1}} \rightarrow \cdots\cdots \rightarrow u_{\mathrm{ds*2}}$$

$$u_0 \rightarrow u_1 \rightarrow \cdots\cdots \rightarrow u_{\mathrm{s*2}}$$

$$u_{\mathrm{d0}} \rightarrow u_{\mathrm{d1}} \rightarrow \cdots\cdots\cdots\cdots \rightarrow u_{\mathrm{ds*3}}$$

$$u_0 \rightarrow u_1 \rightarrow \cdots\cdots\cdots\cdots \rightarrow u_{\mathrm{s*3}}$$

Figure 4-8: Increasing simulation time (from top to bottom) to stabilize the solution.

Using this approach, the instability was avoided, and this can be seen in the adjoint calculation where the adjoint value is representative of the data misfit. In the Figures below we see a sample for the adjoint calculation, shown on the right figure, for a single time step starting from observation prediction and the difference $(u - u_d)$.

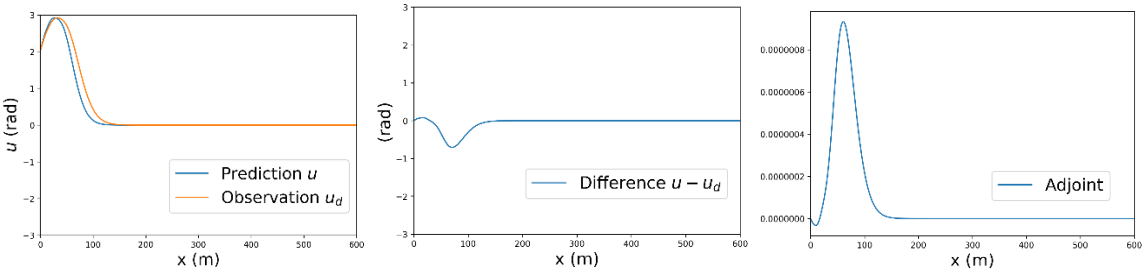


Figure 4-8: Sample Adjoint Calculation

Chapter 5: Conclusion

In this study we described a method to calibrate a time dependent PDE based torsional vibration model using an inverse problem approach. The vibration behavior was modeled in the continuous domain as a one-dimensional wave PDE with proper initial and boundary conditions for the problem. A finite element method and Newmark-beta method were used to discretize the PDE in spatial and time domains. Numerical simulations were made under different conditions of stiffness/damping parameters and input frequencies to generate synthetic data and confirm system behavior.

The inverse problem was formulated using least squares data misfit between observed and predicted displacements. Minimization of the objective function was done through gradient descent and a line search algorithm where parameters are found by iteratively updating the values in the direction of greatest descent. An adjoint method was used to provide a computationally efficiently way to calculate the gradients. The procedure was done to recover stiffness and damping coefficients. Non-linear behavior of the objective function was noticed which caused a very slow convergence rate as the parameters converge to the solution.

The results demonstrate the feasibility of the proposed method to calibrate models using measurement data. The analysis was done to find one unknown parameter at a time, but this method can be expanded to find more than one unknown parameter, although this may require more complex formulation of the inverse problem and larger computational cost.

This model can be implemented in real-time control systems connected to top drive motors to reduce torsional vibration. While this model does not explicitly specify non-linear interaction, the system will be able to approximate the baseline true behavior of the drill-string. If the system diverges from the observed behavior, the parameters can be updated more frequently using the inverse method described in order to improve the match between predicted and actual response.

Further work in this research topic can be done for the following topics:

- Investigate higher dimensional models and methods to incorporate wellbore geometry into the model.
- Investigate the effect of coupling of torsional and axial vibrations on the inverse model to see if improved accuracy can be achieved.
- Confirm accuracy of procedure when fewer observation points are used.
- Investigate effect of non-linear interactions using non-linear boundary condition and friction coefficients.
- Use higher order optimization methods such as conjugate gradient to achieve faster convergence rates.
- Investigate using misfit between acceleration measurements between prediction and observation instead of displacement misfit. This may result in a more stable solution for stiffness inversion by reducing the effect of non-linear behavior of displacement misfit.

References

Ahmadian, H., Nazari, S., Jalali, H. 2007. *Drill String Vibration Modeling Including Coupling Effects.* International Journal of Engineering Science.

Zheng, J., Yang , J., Yang , P. 2017. *Random Fatigue Analysis of Drill-Strings in Frequency Domain.* MATEC Web of Conferences 95.

Navarro-López, E. M., Cortés, D. 2007. *Sliding-mode control of a multi-DOF oilwell drillstring with stick-slip oscillations.* 2007 American Control Conference.

Navarro-López, E. M. 2010. *Bit-Sticking Phenomena in Multidegree-of-freedom Controlled Drillstring.* Exploration & Production vol. 8.

Yigit, A., Christoforou, A. 1998. *Coupled Torsional and Bending Vibrations of Drillstrings Subject to Impact With Friction*. Journal of Sound and Vibrations, vol. 215.

Ghasemloonia, A., Rideout, G., Butt, S. 2015. *A review of drillstring vibration modeling and suppression methods.* Journal of Petroleum Science and Engineering.

Saldivar, B., Mondié, S. *et al.* 2013. *Suppressing Axial-Torsional Coupled Vibrations in Drillstrings.*

Khuleif, Y., Al-Sulaiman, F., Bashmal, S. 2008. *Vibration analysis of drillstrings with string–borehole interaction*. Journal of Mechanical Engineering Science.

Javanmardi, K., Gaspard, D. 1992. *Application of Soft-Torque Rotary Table in Mobile Bay*. IADC/SPE-23913.

Runia, D.J., Dwars, S. 2013. *A Brief History of the Shell "Soft-Torque Rotary System" and Some Recent Case Studies*. IADC/SPE-163548.

Alnaes, M. S., Blechat, J., Hake, J. 2015. The FEniCS Project Version 1.5. Archive of Numerical Software, vol. 3.

COMSOL. 2016. The Finite Element Method (FEM). From website: https://www.comsol.com/multiphysics/finite-element-method

Johnson, S. G. 2012. *Notes on Adjoint Methods for 18.335*.

Dantas, E., Cunha, A. et al. 2019. *An Inverse Problem Via Cross-Entropy Method For Calibration Of A Drill String Torsional Dynamic Model.* 10.26678/ABCM.COBEM2019.COB2019-2216.

Gladwell G.M.L. 2005. *Inverse Problems in Vibration*. Solid mechanics and its applications, vol. 119.

Langtangen, H. P., Madral, K. 2019. Introduction to Numerical Methods for Variational Problems.

Schöberl, J. 2016. Time-stepping methods for wave equations.

Lindfield, G., Penny, J. 2019. Numerical Methods (Fourth Edition).

Demanet, L. 2015. 18.325 Topics in Applied Mathematics: Waves and Imaging. Massachusetts Institute of Technology: MIT OpenCourseWare.

Plessix, R.-E. 2006. A review of the adjoint-state method for computing the gradient of a functional with geophysical applications. Geophysical Journal International, vol. 167.